Continuous Monitoring on Docker with ELK Stack.

Course-end Project 2

CREATED BY LETSATSI MATLALA

Description

Project Objective:

CM on Docker with ELK stack is the process that helps developers to monitor the application in real-time using Kibana.

Sprint Tasks:

1. Setup Docker Environment:

- Install Docker on the target machine.
- Ensure Docker is running and accessible.

2. Create Docker Compose File:

- Define a `docker-compose.yml` file to orchestrate the deployment of ELK stack components and the Spring Boot application.
- Include service definitions for Elasticsearch, Logstash, Kibana, and the Spring Boot application.
- Specify necessary configurations such as network settings, volumes, and environment variables.

3. Elasticsearch Configuration:

- Configure Elasticsearch settings in the `docker-compose.yml`.
- Set up indices and mappings for storing Spring Boot application logs.

4. Logstash Configuration:

- Configure Logstash to ingest logs from the Spring Boot application.
- Define Logstash pipeline to parse and send logs to Elasticsearch.

5. Kibana Configuration:

- Configure Kibana settings in the `docker-compose.yml`.
- Connect Kibana to Elasticsearch.
- Create index patterns in Kibana to visualize Spring Boot application logs.

6. Spring Boot Application Integration:

- Update the Spring Boot application to generate logs.
- Ensure the application logs are in a format compatible with the Logstash configuration.

7. Docker Image Creation:

- Create Docker images for the Spring Boot application, Logstash, Elasticsearch, and Kibana.
- Update the `docker-compose.yml` to use the created images.

8. Integration Testing:

- Deploy the Docker containers locally using the Docker Compose file.
- Verify that logs from the Spring Boot application are successfully indexed in Elasticsearch.
- Confirm the visualization of logs in Kibana.

9. Documentation:

- Document the setup process, including any configuration details and troubleshooting steps.
- Provide instructions for scaling or customizing the monitoring system if needed.

10. Deployment and Production Readiness:

- Test the deployment on a staging environment.
- Address any issues or optimizations identified during testing.
- Ensure proper security configurations for the production environment.

Sprint Review:

- Demonstrate the real-time monitoring capability through Kibana.
- Discuss any challenges faced and how they were resolved.
- Ensure documentation is comprehensive for future reference.

Sprint Retrospective:

- Identify areas for improvement in the process.
- Discuss what went well and what could be improved.
- Plan for the next sprint and any adjustments to the project timeline.

Remember to adapt the tasks based on the specifics of your project and the requirements of your team. Good luck with your project!

THE BELOW TOOLS USED TO CREATE THIS PROJECT:

● Docker

● Docker Compose

● Elasticsearch

● Logstash

● Kibana

● Spring Boot application